

## Longest In-Decreasing Subsequence

Longest In-Decreasing Subsequence คือ ลำดับย่อยที่ยาวที่สุดจากลำดับที่กำหนดให้ โดยสมาชิกในลำดับย่อยดังกล่าวจะพิจารณาจาก 3 กรณี คือ

- 1) สมาชิกในลำดับย่อยเริ่มจากลำดับที่เพิ่มขึ้น (Increasing order) แล้วตามด้วยลำดับที่ลดลง (Decreasing order) หรือ
- 2) ลำดับที่เพิ่มขึ้น หรือ
- 3) ลำดับที่ลดลง

**Input:** บรรทัดที่ 1 ขนาดของลำดับ (n) โดยที่  $1 \leq n \leq 1,000,000$

บรรทัดที่ 2 คือ ลำดับที่ประกอบด้วยจำนวนเต็ม n จำนวน โดยเว้นวรรคข้อมูลแต่ละตัว

**Output:** ขนาดของ Longest In-Decreasing Subsequence

**Example:**

Input	Output	Longest In-Decreasing Subsequence
9 14 12 15 19 17 16 20 13 11	7	14 15 19 17 16 13 11
5 1 9 3 4 5	4	1 3 4 5
5 5 4 3 9 1	4	5 4 3 1

## SendMap

ระหว่างการทำสงครามของ 2 คือประเทศ R และ U ประเทศ U เสียเมือง MP ซึ่งเป็นท่าสำคัญในให้กับประเทศ R ทำให้ประเทศ U ต้องการส่งหน่วยรบพิเศษเข้าไปช่วยเหลือประชาชนที่หลบภัย ณ. จุดหลบภัยซึ่งมีอยู่หลายแห่งในเมือง MP อย่างไรก็ตามตำแหน่งของจุดหลบภัยเหล่านี้เป็นความลับและมีเพียงผู้ว่าการเมือง MP ที่ทราบตำแหน่งของจุดหลบภัยเหล่านั้น

ผู้ว่าการเมือง MP ต้องการส่งแผนที่ขนาด  $M \times N$  (ดูรูปที่ 1) ที่ระบุตำแหน่งของจุดหลบภัยให้กับหน่วยรบพิเศษ โดยสัญลักษณ์บนแผนที่มีความหมายดังนี้

**S** หมายถึงตำแหน่งของจุดหลบภัย

**C** หมายถึงตำแหน่งปลอดภัยที่หน่วยรบสามารถเดินทางผ่านได้

**X** หมายถึงสิ่งกีดขวางหรือตำแหน่งอันตรายที่หน่วยรบไม่สามารถเดินทางผ่านได้

หน่วยรบพิเศษสามารถเดินทางได้ 4 ทิศ คือ ขึ้น, ลง, ซ้าย และขวา เท่านั้น

พิกัดแนวแกน X

จุดเริ่ม

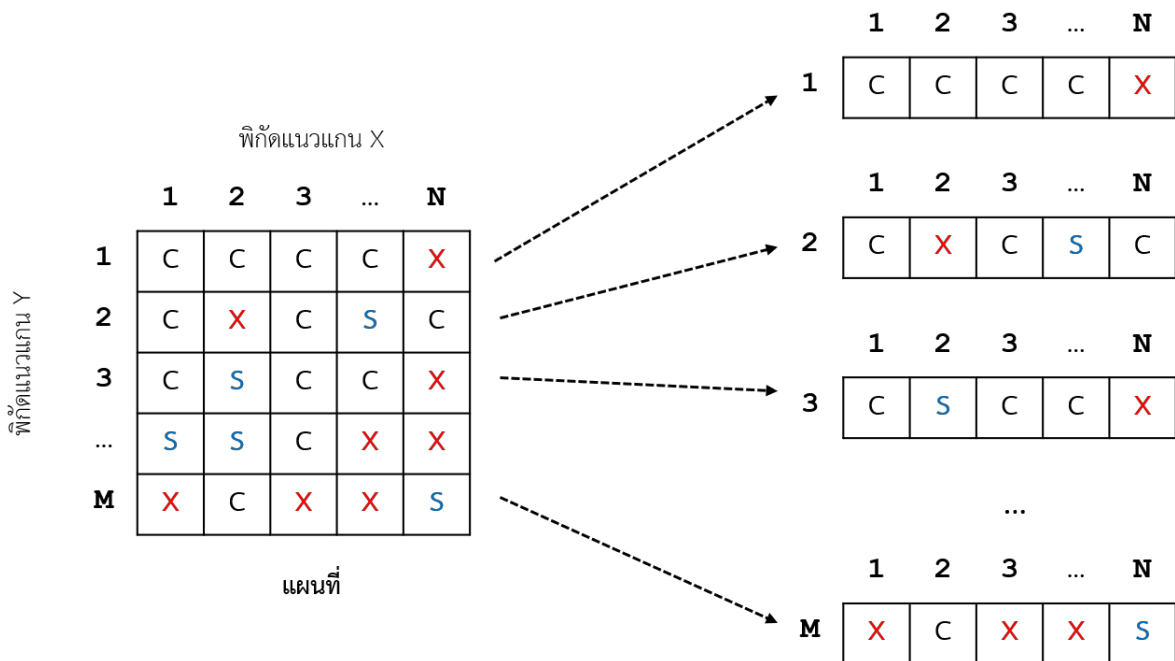
	1	2	3	...	N
1	C	C	C	C	X
2	C	X	C	S	C
3	C	S	C	C	X
...	S	S	C	X	X
M	X	C	X	X	S

พิกัดแนวแกน Y

แผนที่

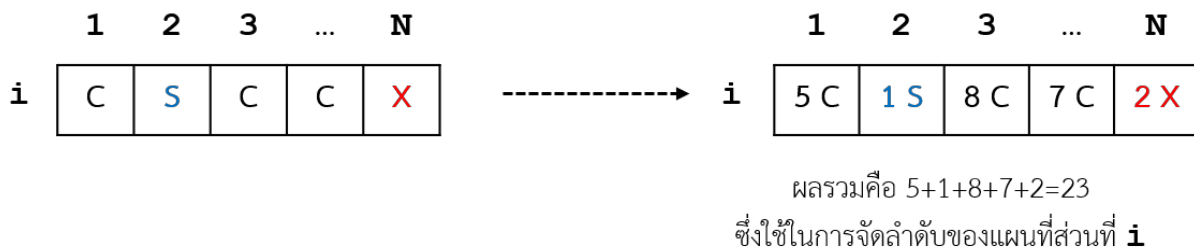
รูปที่ 1

การส่งแผนที่ต้องทำผ่านระบบเครือข่ายที่อาจไม่ปลอดภัย ผู้ว่าการเมือง MP จึงมีความกังวลว่าข้อมูลจะถูกดักจับโดยประเทศ R ทำให้เป็นอันตรายกับประชาชนที่หลบภัยอยู่ ผู้ว่าการเมือง MP จึงทำการแบ่งแผนที่ออกเป็น  $M$  ส่วน ดังแสดงในรูปที่ 2 และทำการส่งแต่ละส่วนผ่านหลายช่องทางการสื่อสารไปยังหน่วยรบพิเศษ



รูปที่ 2

อย่างไรก็ตามเมื่อ  $M$  ส่วนของแผนที่เดินทางไปถึงหน่วยรบพิเศษแล้ว หน่วยรบพิเศษจะต้องทำการประกอบ  $M$  ส่วนของแผนที่เพื่อให้ได้แผนที่ฉบับเต็ม ลำดับของการประกอบแผนที่ถูกเข้ารหัสโดยผู้ว่าการเมือง MP วิธีการเข้ารหัสคือในแต่ละตำแหน่งของแผนที่ จะมีการเพิ่มจำนวนเต็มที่มีค่า 1 ถึง  $M \times N$  ที่ไม่ซ้ำกันเข้าไปด้วย ลำดับในการเรียงแผนที่จะถูกกำหนดโดยผลรวมของจำนวนเต็มทั้งหมดที่บรรจุในแต่ละส่วนของแผนที่ โดยเรียงจากผลรมน้อยที่สุดไปยังมากที่สุด (ดูรูปที่ 3) และไม่มีผลรวมของแผนที่ส่วนใดที่ซ้ำกัน



รูปที่ 3

จงเขียนโปรแกรมเพื่อหาผลรวมของระยะทางจากจุดเริ่มต้นไปยังจุดปลายทางทุกจุดที่หน่วยรบพิเศษสามารถเดินทางไปถึงได้ และหาจำนวนจุดปลายทางที่หน่วยรบพิเศษไม่สามารถเข้าถึงได้ โดยหน่วยรบพิเศษจะเริ่มต้นเดินทางจากพิกัดแกน X เป็น 1 และ พิกัดแกน Y เป็น 1 และที่ตำแหน่งของพิกัดนี้จะมีสัญลักษณ์เป็น C เสมอ

### ข้อมูลนำเข้า (Input)

จำนวน  $M+1$  บรรทัด

บรรทัดแรกเป็นจำนวน  $M$  และ  $N$  ที่ระบุขนาดของแผนที่ ( $3 \leq M, N \leq 1000$ )

บรรทัดที่ 2 ถึง  $M+1$  แต่ละบรรทัดมีจำนวนเต็ม  $N$  ตัว และตัวอักษร  $N$  ตัว สลับกันไป โดยที่

- จำนวนเต็ม แต่มีตัวมีค่าตั้งแต่ 1 ถึง  $M \times N$  และไม่มีค่าที่ซ้ำกัน
- ตัวอักษร แต่ละตัวเป็น S, C หรือ X เท่านั้น

(บรรทัดที่ 2 ถึง  $M+1$  เรียงตามลำดับที่หน่วยรบพิเศษได้รับข้อมูลผ่านช่องทางการสื่อสารต่าง ๆ)

### ข้อมูลส่งออก (Output)

จำนวนเต็ม 2 ตัวคือ

(1) ผลรวมของระยะทางจากจุดเริ่มต้นไปยังจุดหลบภัยทุกจุดที่หน่วยรบพิเศษสามารถเดินทางไปถึงได้

(2) จำนวนจุดหลบภัยที่หน่วยรบพิเศษไม่สามารถเข้าถึงได้

แสดงผลคั่นด้วยช่องว่าง 1 ช่อง

#### ตัวอย่างที่ 1

Input	Output
3 3 2 C 7 C 5 C 4 S 3 X 9 C 8 C 6 S 1 X	4 0

#### ตัวอย่างที่ 2

Input	Output
4 4 13 C 6 C 12 S 3 S 5 C 7 C 9 C 8 C 14 C 2 C 16 C 15 X 11 C 1 C 4 C 10 X	9 0

### ตัวอย่างที่ 3

Input	Output
6 6 22 X 14 X 31 X 34 X 33 X 6 C 26 C 11 X 12 X 4 C 19 X 20 S 27 X 28 X 35 X 2 C 25 X 8 X 21 C 23 X 9 X 32 X 15 X 3 C 7 C 13 X 30 X 36 X 1 X 10 C 24 S 29 C 16 X 5 X 18 X 17 X	3 1

### คำแนะนำ

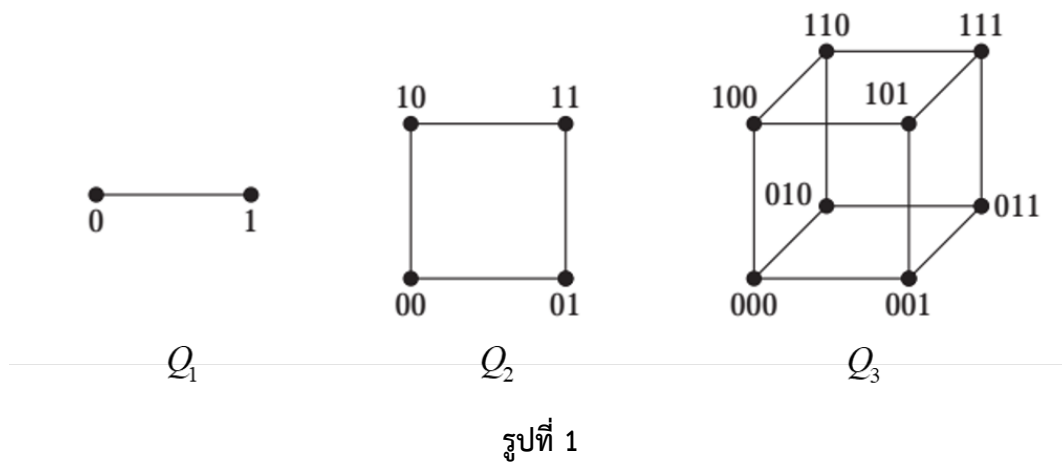
สำหรับข้อนี้

- การเก็บข้อมูลใน vector อาจทำให้เขียนโปรแกรมได้ง่ายกว่าการเก็บด้วยอาร์เรย์
- ประกาศโครงสร้างข้อมูลที่มีจำนวนคอลัมน์เป็น  $N + 1$  เพื่อใช้คอลัมน์สุดท้ายในการเก็บผลรวมของแต่ละแถวที่สามารถคำนวณพร้อมกับการรับข้อมูลได้

## n-Cubes

$n$ -Cubes หรือ  $n$ -dimensional hypercube เขียนแทนด้วย  $Q_n$  เป็นกราฟอย่างง่ายที่มีจุดยอด  $2^n$  จุดยอด (เมื่อ  $n \geq 1$ ) แต่ละจุดยอดแทนด้วยบิตสตริงความยาว  $n$  และมีเส้นเชื่อม  $\left(\frac{n}{2}\right)2^n$  เส้น

จุดยอด 2 จุดยอดของ  $Q_n$  จะมีเส้นเชื่อมก็ต่อเมื่อบิตสตริงที่แทนจุดยอดทั้งสองมีค่าบิตที่ต่างกันเพียง 1 ตำแหน่ง ตัวอย่างของ  $Q_1, Q_2$  และ  $Q_3$  แสดงในรูปที่ 1



จงเขียนโปรแกรมเพื่อแสดงเส้นเชื่อมทั้งหมดของ  $Q_n$  แสดงผลเรียงตามลำดับ lexicographic

ข้อมูลนำเข้า (Input)

จำนวน 1 บรรทัด คือ  $n$  ( $1 \leq n \leq 12$ )

ข้อมูลส่งออก (Output)

จำนวน  $\left(\frac{n}{2}\right)2^n$  บรรทัด แต่ละบรรทัดแสดงเส้นเชื่อมของ  $Q_n$  เรียงตามลำดับ lexicographic

ตัวอย่างที่ 1

Input	Output
1	0-1

ตัวอย่างที่ 2

Input	Output
2	00-01 00-10 01-11 10-11

ตัวอย่างที่ 3

Input	Output
3	000-001 000-010 000-100 001-011 001-101 010-011 010-110 011-111 100-101 100-110 101-111 110-111

## ลูกอม (Candies)

ในวันก่อนปิดภาคการศึกษา มีการแสดงของนักเรียนอนุบาล คุณครูอลิษาต้องการซื้อลูกอมมาแจกนักเรียนแต่ละคนที่ได้รับคะแนนจากการแสดง โดยกำหนดลำดับคะแนนตามการนั่งของนักเรียน และคุณครูอลิษามีเกณฑ์การแจกลูกอมดังนี้

- 1) นักเรียนทุกคนต้องได้ลูกอมอย่างน้อย 1 เม็ด
- 2) นักเรียนที่นั่งติดกัน คนที่ได้คะแนนมากกว่าจะได้รับลูกอมมากกว่านักเรียนคนข้าง ๆ จำนวน 1 เม็ด
- 3) นักเรียนที่นั่งติดกันคะแนนเท่ากัน คนที่อยู่ทางด้านซ้ายมือจะได้มากกว่า เช่น

นักเรียนจำนวน 3 คนมีลำดับคะแนนเป็น 1, 2, 2

ลำดับจำนวนลูกอมที่อลิษาต้องแจก คือ 1, 2, 1

**ข้อมูลนำเข้า** บรรทัดที่ 1 จำนวนเต็ม  $N$  แทนจำนวนนักเรียน โดยที่  $1 \leq N \leq 10^5$

บรรทัดที่ 2 ถึงบรรทัดที่  $N + 1$  จำนวนเต็ม  $N$  จำนวนแทนคะแนนของนักเรียนแต่ละคนเรียงตามตำแหน่งที่นั่ง โดยที่  $1 \leq \text{คะแนน} \leq 10^5$

**ข้อมูลนำออก** แสดงจำนวนลูกอมที่คุณครูอลิษาต้องซื้ออย่างน้อยที่สุดเพื่อให้เพียงพอกับจำนวนนักเรียนตามเงื่อนไขที่กำหนด

ตัวอย่าง

Input	Output	คำอธิบาย
3 1 2 2	4	ลำดับจำนวนลูกอมที่อลิษาต้องแจก คือ 1 2 1 ดังนั้นจำนวนลูกอมทั้งหมดคือ $1+2+1 = 4$
10 2 4 2 6 1 7 8 9 2 1	19	ลำดับจำนวนลูกอมที่อลิษาต้องแจก คือ 1 2 1 2 1 2 3 4 2 1 ดังนั้นจำนวนลูกอมทั้งหมดคือ $1+2+1+2+1+2+3+4+2+1 = 19$
8 2 4 3 5 2 6 4 5	12	ลำดับจำนวนลูกอมที่อลิษาต้องแจก คือ 1 2 1 2 1 2 1 2 ดังนั้นจำนวนลูกอมทั้งหมดคือ $1+2+1+2+1+2+1+2 = 12$